

Batch Computing Support

C. Preuss¹ and K. Schwarz¹

¹GSI, Darmstadt, Germany

Current State

After getting good experiences with GridEngine [1] on our (Icarus) cluster in 2011 [2], we decided to use the same technique also for the new larger cluster (Prometheus), which was delivered and installed in spring 2012. The compute clusters and their properties are listed in table 1.

With now two additional clusters and the phase-out of Debian Etch at GSI, we decided to shutdown the old LSF cluster end of 2012. Due to the circumstances that we have two clusters with GE not differing only in the size but also in the used Lustre instances, CPUs/memory, location, and network (and a new Debian version) the movement of the last users from LSF to GE had taken more time than expected, but was in the end successful.

Cluster	Icarus	Prometheus
Cores	1600	9000
Cores/Node	16	24
Memory	2GB/core	2.7GB/core
Lustre instance	/lustre	/hera
Network	Ethernet	InfiniBand
Location	BG2	MiniCube

Table 1: The compute clusters at GSI and their properties.

BlackHole detection script

As the batch farm has become larger over the years, the demand for a tool which can identify problematic nodes automatically came up. The decision was made to develop a so called "BlackHole" detector script which monitors the job flow on the batch nodes, identifies anomalies and decides whether an alert is raised or not.

During the test phase we have identified several show stoppers preventing us from using it as a fully automated system, so currently it is in an adjustment and configuration phase. A proper balance between finding a good and fast automatic detection and not too many blind alerts still needs to be found.

Accounting and Visualization Tool

As some experiments are curious about their resource usage we decided to develop an accounting tool for the batch farm, which was done by one of our KoSI students, Sinisa Jovanovic. He will describe his work in another article [3], "A Visualisation Tool for Accounting Data".

Outview

As we are still using the last GE version developed by SUN (6.2.5) the problems described in the Scientific Report from last year (memory restrictions, MPI scheduling) are still existing.

Luckily the development of GE is now ongoing, but unfortunately in several different branches.

So there are now OracleGE and UnivaGE as commercial versions, and two open source trunks, OpenGE and SonOfGE [4].

As one reason for using GE was that it is for free we will stay with the open source trunks of GE, favoured is here SonOfGE as it seems to be more alive than OpenGE.

A test setup will be deployed during the next months, together with a database for the accounting data, a mechanism for shutting down unused batch nodes and the new Debian Wheezy as operating system.

References

- [1] web site of gridengine:
<http://gridengine.org>
- [2] Wiki page with GridEngine Documentation:
<http://wiki.gsi.de/cgi-bin/view/Linux/GridEngineUsage>
- [3] S. Jovanovic, C. Preuss, K. Schwarz, "A Visualisation Tool for Accounting Data", this report.
- [4] web site of SGE:
<http://arc.liv.ac.uk/SGE>